

Deformable 3d shape retrieval using a spectral geometric descriptor

Waleed Mohamed¹ · A. Ben Hamza¹

© Springer Science+Business Media New York 2016

Abstract In this paper, we propose a deformable 3D shape matching and retrieval approach using a spectral skeleton that encodes nonrigid object structures. This spectral skeleton is constructed from the second eigenfunction of the Laplace-Beltrami operator defined on the surface of a 3D shape, and thus it is invariant to isometric transformations. In addition to its intrinsic property, our proposed shape descriptor is compact, robust to noise, discriminative, and efficient to compute. We also present a graph matching framework by comparing the shortest paths between skeleton endpoints. Extensive experimental results demonstrate the feasibility of the proposed shape retrieval approach on three standard benchmarks of nonrigid 3D shapes.

Keywords Spectral reeb graph · Laplace-Beltrami · Shortest path · Retrieval

1 Introduction

In light of the latest software, hardware and computing advancements, 3D technology has grown beyond being a buzzword. Today, 3D technology has become an essential part of the modern lifestyle and is gaining momentum rapidly, from consumer demand for in-home 3D television

experiences to far-reaching positive implications for health-care through the use of advanced 3D medical imaging systems aimed at improving patient outcomes and expanding their clinical practice.

The importance of 3D shape recognition is erupting due to the difficulty in processing information expeditiously without its recognition. With the increasing use of 3D scanners and as a result of emerging multimedia computing technologies, vast databases of 3D models are distributed freely or commercially on the World Wide Web. The availability and widespread usage of such large databases, coupled with the need to explore 3D models in depth as well as in breadth, has sparked the need to organize and search these vast data collections, retrieve the most relevant selections, and permit them to be effectively reused. 3D objects consist of geometric and topological information, and their compact representation is an important step towards a variety of computer vision applications, particularly matching and retrieval in a database of 3D models. The first step in 3D object matching usually involves finding a reliable shape descriptor or skeletal graph, which will encode efficiently the 3D shape information. Skeletonization aims at reducing the dimensionality of a 3D shape while preserving its topology [1, 2]. Unlike text documents, 3D models are not easily retrieved due largely to the variability of their shapes. Attempting to retrieve a 3D model using textual annotation and a conventional text-based search engine would not work properly in many cases [4]. The annotations added by users depend on various factors, including language, culture, age, and gender. In contrast, content-based 3D shape retrieval methods, which typically use the shape properties of the query shape to search for similar models, perform better than text-based approaches [4].

✉ A. Ben Hamza
hamza@ciise.concordia.ca

¹ Concordia Institute for Information Systems Engineering,
Concordia University, Montreal, QC, Canada

The vast majority of 3D shape representation techniques proposed in the literature of computer graphics and computer vision are primarily based on geometric and topological representations which represent the features of an object [5–7]. For example, Siddiqi et al. [5] introduced a shock detection approach based on singularity theory to generate a skeletal shape model. Also, Siddiqi et al. [8] proposed a directed acyclic graph representation for 3D retrieval using medial surfaces. This approach uses the geometric information associated with each graph node along with an eigenvalue labeling of the adjacency matrix of the subgraph rooted at that node. Cornea et al. [9] devised a 3D matching framework for 3D volumetric objects using a many-to-many matching algorithm. This algorithm is based on establishing correspondences among two skeletal representations via distribution-based matching in metric spaces. Hassouna et al. [10] proposed a level set based framework for robust centerline extraction of 2D shapes and 3D volumetric objects. This approach is based on the gradient vector flow and uses a wave propagation technique, which identifies the curve skeletons as the wave points of maximum positive curvatures. Tagliasacchi et al. [11] introduced a curve skeleton extraction algorithm from imperfect point clouds. A major drawback of curve skeletons is that they cannot capture general shape features such as surface ridges, and are essentially restricted to objects which resemble connected tubular forms.

An alternative to feature-based representations are global methods, which represent a 3D object by a global measure or shape distribution defined on the surface of the object [12–14, 16]. Ankerst et al. [12] used shape histograms to analyze the similarity of 3D molecular surfaces. These histograms are built from uniformly distributed surface points taken from the molecular surfaces, and are defined on concentric shells and sectors around the centroid of the surface. Osada et al. [13] proposed a global approach for computing shape signatures of arbitrary 3D models. The key idea is to represent an object by a global histogram based on the Euclidean distance defined on the surface of an object. More recently, Ion et al. [17] presented an articulation-insensitive shape matching approach by constructing histograms from the eccentricity transform using geodesic distances. Kazhdan et al. [14] proposed a rotation invariant spherical harmonic representation that transforms rotation dependent shape descriptors into rotation independent ones. Chen et al. [18] presented a lightfield descriptor for 3D object retrieval by comparing ten silhouettes of the 3D shape obtained from ten viewing angles distributed uniformly on the viewing enclosing sphere. The dissimilarity between two shapes is computed as the minimal distance obtained by rotating the viewing sphere of one lightfield descriptor relative to the

other lightfield descriptor. The computation of this descriptor is, however, significantly time consuming compared to spherical harmonics [19].

In this paper, we propose a spectral skeletonization approach that aims at representing 3D objects with topological coding, which we refer to as *Spectral Reeb Graph* (SRG). Topology represents the connectedness of a shape and enables parts of shapes, which are connected, to be mapped and drawn equivalently. One of the key mathematical tools used to study the topology of spaces is Morse theory, which is the study of the relationship between functions on a space and the shape of the space. Morse theory studies the properties of a Morse function which has only nondegenerate singular points [2, 3], and it describes the topology changes of the level sets of this function at those singularities. Regular or noncritical points do not affect the number or genus of the components of the level sets. It can be shown that Morse functions are dense and stable in the set of all smooth functions, that is the structure of nondegenerate singularities does not change under small perturbations [2, 3]. A Morse theoretic representation that captures topological properties of objects is the so-called *Reeb graph* proposed in [7], which is based on the Morse height function. The vertices of the Reeb graph are the singular points of a Morse function defined on the surface of a 3D object [2, 7]. The height function-based approach may lead to the extraction of an unbounded number of critical points, except in the case of triangle meshes where the number of critical points is bounded by the number of mesh vertices. This limitation has been addressed in [20] by introducing a fair Morse function that produces the least possible number of critical points. Since the level sets of the height function are horizontal planes perpendicular to the height axis, the main weakness of such Reeb graphs is that they are not invariant to rotation. Hilaga et al. [6] used the geodesic distance from point to point on a surface to overcome the problem of automatic extraction of the source point. The geodesic integral is, however, computed using a selected (typically small) random subset of points on the surface, which may lead to inaccuracies in terms of effectively capturing the topological structure of the surface. Moreover, another disadvantage of using the geodesic distance is its sensitivity to topological changes. That is, modifying the shape connectivity may significantly alter the shortest paths between feature points, resulting in significant changes of the geodesic distance. Tierny et al. [21] presented a structural oriented Reeb graph based method for partial 3D shape retrieval. Partial similarity between two shapes is then evaluated by computing a variant of their maximum common sub-graph. Aouada et al. [22] proposed a topological Reeb graph using an intrinsic global geodesic function defined on the surface of

a 3D object. This approach decomposes a shape into primitives, and then detailed geometric information is added by tracking the evolution of Morse's function level curves along each primitive. A detailed overview of the mathematical properties of Reeb graphs and their applications to shape analysis is presented in [23]. Pascucci et al. [24] introduced a robust method for fast Reeb graph computation that is able to handle non-manifold meshes. Also, Patane et al. [25] proposed an efficient Reeb graph computation algorithm by studying the evolution of the level sets only at the saddle points of a Morse function.

More recently, there has been a surge of interest in the spectral analysis of the Laplace-Beltrami (LB) operator, resulting in a slew of applications to manifold learning [26], object recognition and shape analysis [27–30], and multimedia protection [31]. It is worth pointing out that spherical harmonics [14] are nothing but the LB eigenfunctions on the sphere. Reuter [28] introduced a Morse-theoretic method for shape segmentation and registration using the topological features of the LB eigenfunctions. These eigenfunctions are computed via a cubic finite element method on triangular meshes, and are arranged in increasing order of their associated eigenvalues. Rustamov [29] proposed a feature descriptor referred to as the global point signature (GPS), which is a vector whose components are scaled eigenfunctions of the LBO evaluated at each surface point. GPS is invariant under isometric deformations of the shape, but it suffers from the problem of eigenfunctions switching whenever the associated eigenvalues are close to each other. Bronstein et al. [30] proposed a non-rigid shape retrieval approach using bags of features based on the heat kernel signature (HKS) [32]. HKS is a temporal shape descriptor, which is defined as an exponentially-weighted combination of the eigenfunctions of the LBO on a manifold. HKS is a local shape descriptor that has a number of desirable properties, including robustness to small perturbations of the shape, efficiency, and invariance to isometric transformations. However, HKS depends on the time parameter, which needs to be set a priori. Also, the discrete heat kernel requires finding a few hundred eigenvalues and eigenfunctions of a typically large LB matrix, which is often a computationally expensive process. In addition, the choice of the vocabulary size, the time parameter, and the number of eigenvalues/eigenfunctions can have an impact on the performance of the HKS-based retrieval algorithm. The idea of HKS was independently proposed by Gébal et al. [33] for 3D shape skeletonization and segmentation under the name of *auto diffusion function* (ADF). As the name suggests, ADF describes the diffusion from a surface point to itself. Shi et al. [34] used the level curves of the second eigenfunction of the LBO to construct the spectral skeleton

of 3D neuroanatomical structures. In addition to having a nice geometric property of following the pattern of the overall shape of a 3D object, the second eigenfunction of the LBO can capture the intrinsic structure of elongated shapes (e.g. hippocampus) and it is also invariant to isometric transformations. Moreover, the spectral skeleton is invariant to the pose of the shape [34]. More recent spectral signatures include the wave kernel signature [35] and the spectral graph wavelet signature [36, 37]. A comprehensive list of global spectral descriptors can be found at [38].

Motivated by the aforementioned invariance properties of the second eigenfunction of the LBO, we propose to use the spectral Reeb graph framework to construct the shape skeleton of a 3D object. The key idea is to identify and encode regions of topological interest of a 3D shape in the Morse-theoretic framework. More precisely, the level sets (isocontours) of the second eigenfunction are computed (identified), and then each level set is encoded as a skeleton node representing the centroid of the isocontour.

The rest of this paper is organized as follows. In Section 2, we provide a brief overview to the Laplace-Beltrami operator (LBO), followed by its spectral analysis in the discrete domain. In Section 3, we explore the connection between Morse theory and the spectrum of the LBO. Then, we introduce an algorithm for computing the spectral Reeb graph of a 3D object based on the second eigenfunction of the LBO. Section 4 introduces the path dissimilarity skeleton graph matching method by comparing the relative shortest paths between the skeleton endpoints. In Section 5, we present experimental results for topological coding using the spectral Reeb graph and we demonstrate the feasibility of this skeletal graph as a shape descriptor for 3D object matching and retrieval. A comprehensive comparison with spectral methods is presented in Section 6. Finally, in Section 7, we summarize our results and point out future work directions.

2 Background

In computer graphics and geometric-aided design, triangle meshes have become the *de facto* standard representation of 3D objects. We model a 3D shape as a triangle mesh \mathbb{M} whose vertices are sampled from a Riemannian manifold. A triangle mesh \mathbb{M} may be defined as a graph $\mathbb{G} = (\mathcal{V}, \mathcal{E})$ or $\mathbb{G} = (\mathcal{V}, \mathcal{T})$, where $\mathcal{V} = \{\mathbf{v}_1, \dots, \mathbf{v}_n\}$ is the set of vertices, $\mathcal{E} = \{e_{ij}\}$ is the set of edges, and $\mathcal{T} = \{\mathbf{t}_1, \dots, \mathbf{t}_m\}$ is the set of triangles, as depicted in the enlarged view of Fig. 1 (left). Each edge $e_{ij} = [\mathbf{v}_i, \mathbf{v}_j]$ connects a pair of vertices $\{\mathbf{v}_i, \mathbf{v}_j\}$. Two distinct vertices $\mathbf{v}_i, \mathbf{v}_j \in \mathcal{V}$ are adjacent (denoted by $\mathbf{v}_i \sim \mathbf{v}_j$ or simply $i \sim j$) if they are connected by an edge,

i.e., $e_{ij} \in \mathcal{E}$. The neighborhood (also called *ring*) of a vertex \mathbf{v}_i is the set

$$\mathring{\mathbb{V}}_i = \{\mathbf{v}_j \in \mathcal{V} : \mathbf{v}_j \sim \mathbf{v}_i\}. \quad (1)$$

The degree d_i of a vertex \mathbf{v}_i is simply the cardinality of $\mathring{\mathbb{V}}_i$. Figure 1 (right) depicts an example of a neighborhood $\mathring{\mathbb{V}}_i$, where the degree of the vertex \mathbf{v}_i is $d_i = 6$.

2.1 Laplace-Beltrami operator

Let \mathbb{M} be a smooth orientable 2-manifold (surface) embedded in \mathbb{R}^3 . A global parametric representation (embedding) of \mathbb{M} is a smooth vector-valued map \mathbf{x} defined from a connected open set (parametrization domain) $U \subset \mathbb{R}^2$ to $\mathbb{M} \subset \mathbb{R}^3$ such that $\mathbf{x}(\mathbf{u}) = (x^1(\mathbf{u}), x^2(\mathbf{u}), x^3(\mathbf{u}))$, where $\mathbf{u} = (u^1, u^2) \in U$. The parametric representation $\mathbf{x} : U \rightarrow \mathbb{R}^3$ of the surface \mathbb{M} is a map, represented by a vector-valued function $\mathbf{x} = \mathbf{x}(u^1, u^2)$, which associates with each point (u^1, u^2) in U a point \mathbf{x} of \mathbb{M} with position vector $\mathbf{x}(u^1, u^2)$. Thus, we use \mathbf{x} to denote a map as well as a point on the surface.

Let $\mathbf{x} = \mathbf{x}(u^1, u^2) \in \mathbb{M}$ and $f : \mathbb{M} \rightarrow \mathbb{R}$ be a real-valued function defined on a smooth, orientable 2-manifold \mathbb{M} embedded in \mathbb{R}^3 . The function f is differentiable at \mathbf{x} if the composition function $f \circ \mathbf{x} : U \rightarrow \mathbb{R}$ is differentiable at (u^1, u^2) . Given a twice-differentiable function $f : \mathbb{M} \rightarrow \mathbb{R}$, the LBO [47, 54] is defined as

$$\Delta_{\mathbb{M}} f = -\operatorname{div}(\nabla_{\mathbb{M}} f), \quad (2)$$

where $\nabla_{\mathbb{M}} f$ is the intrinsic gradient vector field given by

$$\nabla_{\mathbb{M}} f = \sum_{i,j=1}^2 g^{ij} \frac{\partial f}{\partial u^i} \frac{\partial \mathbf{x}}{\partial u^j} = (\mathbf{x}_1, \mathbf{x}_2) \mathbf{g}^{-1} \nabla f \in \mathbb{R}^3, \quad (3)$$

with

$$\nabla f = \left(\frac{\partial f(\mathbf{x}(u^1, u^2))}{\partial u^1}, \frac{\partial f(\mathbf{x}(u^1, u^2))}{\partial u^2} \right)^T \in \mathbb{R}^2, \quad (4)$$

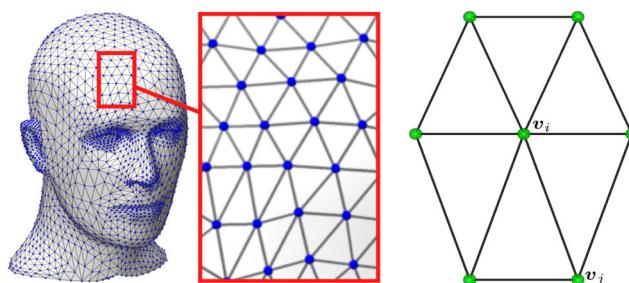


Fig. 1 Triangular mesh representation (left); Vertex neighborhood (right)

and $g^{ij} = \langle du^i, du^j \rangle$ denote the elements of the inverse of the metric tensor $\mathbf{g} = (g_{ij})$, i.e.,

$$\mathbf{g}^{-1} = (g^{ij}) = \frac{1}{\det \mathbf{g}} \begin{pmatrix} g_{22} & -g_{12} \\ -g_{12} & g_{11} \end{pmatrix}. \quad (5)$$

The metric tensor \mathbf{g} is an intrinsic quantity in the sense that it relates to measurements inside the surface.

The divergence of a vector field $\mathbf{F} = F^1 \mathbf{x}_1 + F^2 \mathbf{x}_2$ on the tangent space $T_{\mathbf{x}} \mathbb{M}$ is given by

$$\operatorname{div} \mathbf{F} = \frac{1}{\sqrt{\det \mathbf{g}}} \sum_{i=1}^2 \frac{\partial}{\partial u^i} \left(F^i \sqrt{\det \mathbf{g}} \right). \quad (6)$$

Thus, the LBO may be expressed as

$$\begin{aligned} \Delta_{\mathbb{M}} f &= -\frac{1}{\sqrt{\det \mathbf{g}}} \sum_{i,j=1}^2 \frac{\partial}{\partial u^j} \left(\sqrt{\det \mathbf{g}} g^{ij} \frac{\partial f}{\partial u^i} \right) \\ &= -\sum_{i,j=1}^2 g^{ij} \frac{\partial}{\partial u^j} \frac{\partial f}{\partial u^i} + (\text{lower order terms}). \end{aligned} \quad (7)$$

The LBO is a linear, second-order partial differential operator acting on the space of real-valued functions defined on a manifold. It is a generalization of the Laplace operator to non-Euclidean spaces.

Let $L^2(\mathbb{M})$ be the space of all smooth, square-integrable functions on a manifold \mathbb{M} . It is a Hilbert space endowed with inner product

$$\langle f_1, f_2 \rangle = \int_{\mathbb{M}} f_1(\mathbf{x}) f_2(\mathbf{x}) da(\mathbf{x}), \quad \text{for all } f_1, f_2 \in L^2(\mathbb{M}), \quad (8)$$

where $da(\mathbf{x})$ (or simply dx) denotes the measure from the area element of a Riemannian metric on \mathbb{M} .

Discretization A real-valued function $f : \mathcal{V} \rightarrow \mathbb{R}$ defined on the mesh vertex set may be represented as an n -dimensional vector $\mathbf{f} = (f(i)) \in \mathbb{R}^n$, where the i th component $f(i)$ denotes the function value at the i th vertex in \mathcal{V} .

Using a mixed finite element/finite volume method on triangle meshes [51], the value of $\Delta_{\mathbb{M}} f$ at a vertex \mathbf{v}_i (or simply i) can be approximated using the cotangent weight scheme as follows:

$$\Delta_{\mathbb{M}} f(i) \approx \frac{1}{a_i} \sum_{j \sim i} \frac{\cot \alpha_{ij} + \cot \beta_{ij}}{2} (f(i) - f(j)), \quad (9)$$

where α_{ij} and β_{ij} are the angles $\angle(\mathbf{v}_i \mathbf{v}_{k_1} \mathbf{v}_j)$ and $\angle(\mathbf{v}_i \mathbf{v}_{k_2} \mathbf{v}_j)$ of two faces $\mathbf{t}^\alpha = \{\mathbf{v}_i, \mathbf{v}_j, \mathbf{v}_{k_1}\}$ and $\mathbf{t}^\beta = \{\mathbf{v}_i, \mathbf{v}_j, \mathbf{v}_{k_2}\}$ that are adjacent to the edge $[i, j]$, and a_i is the area of the Voronoi cell (shaded polygon) at vertex i , as shown in Fig. 2. It should be noted that the cotangent weight scheme is numerically consistent and preserves several important properties of the continuous LBO, including symmetry and positive semi-definiteness [49].

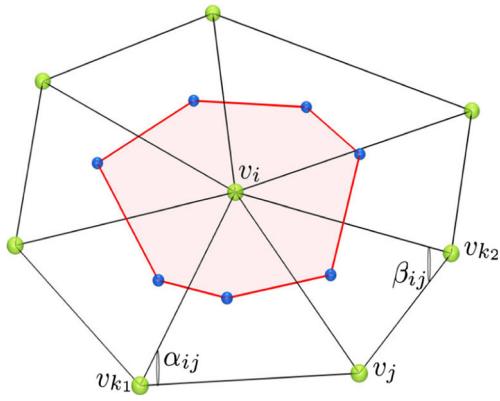


Fig. 2 Cotangent weight scheme: illustration of the angles α_{ij} and β_{ij}

By defining the weight function $w : \mathcal{V} \times \mathcal{V} \rightarrow \mathbb{R}$ as

$$w_{ij} = \begin{cases} \frac{\cot \alpha_{ij} + \cot \beta_{ij}}{2a_i} & \text{if } i \sim j \\ 0 & \text{o.w.,} \end{cases} \quad (10)$$

we may rewrite the LBO given by Eq. 9 as

$$\Delta_{\mathbb{M}} f(i) = \sum_{j \sim i} w_{ij} (f(i) - f(j)), \quad (11)$$

with associated matrix $\mathbf{L} = (\ell_{ij})$ given by

$$\ell_{ij} = \begin{cases} d_i & \text{if } i = j \\ -w_{ij} & \text{if } i \sim j \\ 0 & \text{o.w.} \end{cases} \quad (12)$$

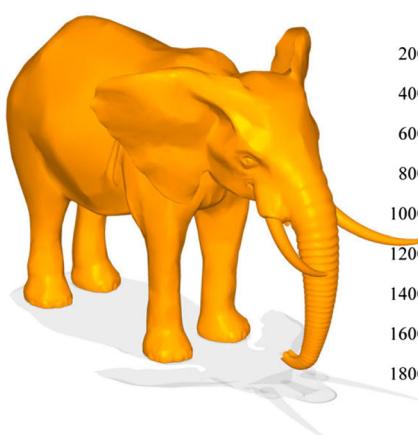
where $d_i = \sum_{j \sim i} w_{ij}$ is the degree of the vertex i . Note that $w_{ij} \neq w_{ji}$ implies \mathbf{L} is not a symmetric matrix.

Spectral analysis Since the matrix \mathbf{L} associated to the LBO is not symmetric, it follows that the spectrum (set of eigenvalues) of the standard matrix eigenvalue problem

$$\mathbf{L}\varphi_\ell = \lambda_\ell \varphi_\ell \quad (13)$$

may not be real [29], where φ_ℓ are the unknown associated eigenfunctions (i.e., eigenvectors which can be thought of as

Fig. 3 3D elephant model and sparsity pattern plot of the associated stiffness matrix \mathbf{W}



functions on the mesh vertices). Noting that $w_{ij} = \eta_{ij}/a_i$, where

$$\eta_{ij} = \begin{cases} \frac{\cot \alpha_{ij} + \cot \beta_{ij}}{2} & \text{if } i \sim j \\ 0 & \text{o.w.} \end{cases} \quad (14)$$

we may factorize the matrix \mathbf{L} as $\mathbf{L} = \mathbf{A}^{-1} \mathbf{W}$, where $\mathbf{A} = \text{diag}(a_i)$ is a positive definite diagonal matrix (often called mass matrix), and $\mathbf{W} = (c_{ij}) = \text{diag}(\sum_{k \neq i} \eta_{ik}) - (\eta_{ij})$ is a sparse symmetric matrix (also referred to as stiffness or conformal Laplacian matrix) whose elements are given by

$$c_{ij} = \begin{cases} \sum_{k=1}^n \eta_{ik} & \text{if } i = j \\ -\eta_{ij} & \text{if } i \sim j \\ 0 & \text{o.w.} \end{cases} \quad (15)$$

Therefore, we may write Eq. 13 as a generalized eigenvalue problem

$$\mathbf{W}\varphi_\ell = \lambda_\ell \mathbf{A}\varphi_\ell, \quad (16)$$

which can be solved efficiently using the Arnoldi method of ARPACK, where $\varphi_\ell = (\varphi_\ell(v_1), \dots, \varphi_\ell(v_n))^T$ is the unknown ℓ th eigenvector (i.e., φ_ℓ is the ℓ th generalized eigenfunction evaluated at n mesh vertices). Figure 3 shows a 3D elephant model and the sparsity pattern of the stiffness matrix \mathbf{W} . Recall that the sparsity pattern (or support) of a matrix $\mathbf{S} = (s_{ij})$ is the set of indices ij with $s_{ij} \neq 0$.

3 Proposed framework

In this section, we give a detailed description of our proposed method that makes use of spectral geometry and Morse theory. Spectral geometry is concerned with the eigenvalue spectrum of the LBO on a compact Riemannian manifold, and aims at describing the relationships between such a spectrum and the geometric structure of the manifold. Spectral geometric problems can be broadly divided into two main categories: direct problems and inverse problems. A direct problem attempts to infer information about the

eigenvalues and eigenfunctions of the LBO from knowledge of the geometry of the manifold. In the inverse problem, however, the goal is to investigate what geometric and topological information of the manifold can be recovered from the spectrum of the LBO. The inverse spectral problem has been translated over the years into the conversational question “Can one hear the shape of a manifold?” by several mathematicians, most notably by Kac [50] because of its analogy with the wave equation, which models the transverse vibrations of ideal stretched objects. The answer to this question is demonstrably untrue: one cannot hear the shape of a manifold. In other words, the spectrum cannot completely determine the geometry and topology of a manifold as there exist non-isometric manifolds with the same spectrum [55]. However, some geometric and topological information about the manifold can be inferred, such as the dimension, volume, curvature and the Euler characteristic.

Morse theory for topological modeling Morse theory explains the presence and the stability of singular points in terms of the topology of the underlying smooth manifold. The basic principle is that the topology of a manifold is very closely related to the singular points of a smooth function defined on that manifold [3]. A smooth function $f : \mathbb{M} \rightarrow \mathbb{R}$ on a smooth manifold \mathbb{M} is called a *Morse function* if all its singular points are nondegenerate, i.e. the Hessian matrix is nonsingular at every singular point. A point \mathbf{x} is called a *regular point* of f if the differential $df : T_{\mathbf{x}}\mathbb{M} \rightarrow \mathbb{R}$ is *surjective*, that is, the Jacobian matrix (3×1 in the case of a 2-manifold) has rank equal to $\dim(\mathbb{R}) = 1$. Otherwise, the point \mathbf{x} is called a *critical point*. The only nondegenerate singularities are the minimum, maximum and saddle points, as depicted in Fig. 4 (left), where the height function $h : \mathbb{M} \rightarrow \mathbb{R}$ is a real-valued function such that $h(x, y, z) = z$ for all $\mathbf{x} = (x, y, z) \in \mathbb{M}$. That is, h is an orthogonal projection with respect to the z -axis. Nondegenerate singularities are isolated, that is, there cannot be a sequence of nondegenerate singularities converging to a nondegenerate singularity $\mathbf{x} \in \mathbb{M}$. A *level set*

$f^{-1}(a)$ of f at a value $a \in \mathbb{R}$ may be composed of one or many connected components. The Morse deformation lemma states that if no critical points exist between two level sets of f , then the two level sets are topologically equivalent and can be deformed onto one another [2]. In particular, they consist of the same number of connected components. Furthermore, Morse theory implies that topological changes on the level sets occur only at critical points. The level sets of the height function are the intersections of the manifold \mathbb{M} with planes orthogonal to the z -axis, as pictured in Fig. 4 (right). The original manifold may then be reconstructed if we know all its sections by these parallel planes (i.e., the surface is the union of these planes). Clearly, the level sets of the height function may have isolated points, or curves, or may contain an open subset of the plane. Furthermore, the level sets may be connected or disconnected, and the curves may have complicated singularities.

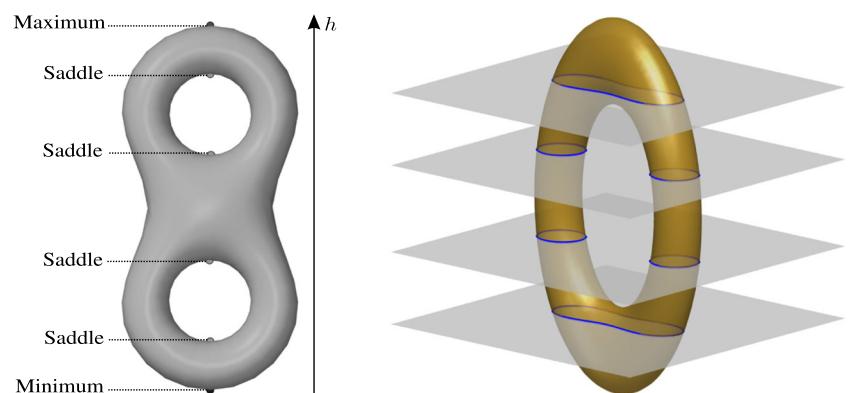
An interesting concept related to Morse theory and very useful to analyze a surface topology is the Reeb graph. The latter is defined as a quotient space \mathbb{M}/\sim with the equivalence relation given by $\mathbf{x} \sim \mathbf{y}$ if and only if $f(\mathbf{x}) = f(\mathbf{y})$ and \mathbf{x}, \mathbf{y} belong to the same connected component of $f^{-1}(f(\mathbf{x}))$. An equivalence class is defined as $[\mathbf{x}] = \{\mathbf{y} \in \mathbb{M} : \mathbf{x} \sim \mathbf{y}\}$. Intuitively, \mathbb{M}/\sim is a space created by taking the space \mathbb{M} and gluing \mathbf{x} to any \mathbf{y} that satisfies $\mathbf{y} \sim \mathbf{x}$. The classes $[\mathbf{x}]$ are the connected components for the Reeb graph, and being in the same component is an equivalence relation:

$$\mathbf{y} \sim \mathbf{x} \iff f(\mathbf{y}) = f(\mathbf{x}) \text{ and } \mathbf{x}, \mathbf{y} \in \mathcal{C}, \quad (17)$$

where \mathcal{C} denotes the connected component of $f^{-1}(f(\mathbf{x}))$.

Proposed algorithm Intuitively, one may think of a Riemannian manifold as a musical instrument together with the musician who plays it. In this setting, the eigenvalues of the LBO correspond to the harmonics of the instrument; they may depend of the musician (i.e. Riemannian metric). Spectral geometry aims at describing the relationships between the musical instrument (i.e. manifold) and the sound it is capable of sending out (spectrum). We may sort the eigenvalues in ascending order as $0 = \lambda_1 < \lambda_2 \leq \dots \leq \lambda_n$

Fig. 4 Critical points of the height function on a double torus. (left); Cross-sections of the height function on a torus (right)



with associated orthonormal eigenfunctions $\varphi_1, \varphi_2, \dots, \varphi_n$, where the orthogonality of the eigenfunctions is defined in terms of the \mathbf{A} -inner product, i.e.,

$$\langle \varphi_k, \varphi_\ell \rangle_{\mathbf{A}} = \sum_{i=1}^n a_i \varphi_k(i) \varphi_\ell(i) = \delta_{k\ell}, \quad \text{for all } k, \ell = 1, \dots, n \quad (18)$$

or equivalently

$$\langle \varphi_k, \varphi_\ell \rangle_{\mathbf{A}} = \varphi_k^T \mathbf{A} \varphi_\ell = \delta_{k\ell}, \quad \text{for all } k, \ell = 1, \dots, n. \quad (19)$$

In a similar fashion, we may rewrite (16) in matrix form as follows:

$$\mathbf{W}\Phi = \mathbf{A}\Phi\Lambda, \quad (20)$$

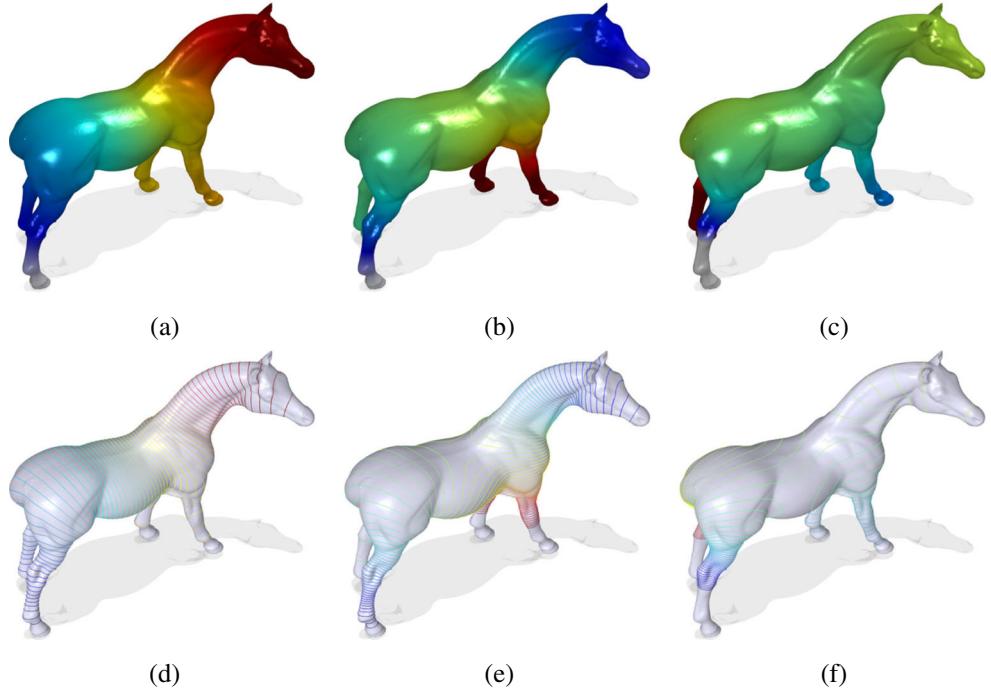
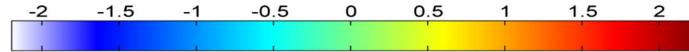
where $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n)$ is an $n \times n$ diagonal matrix with the (generalized) eigenvalues λ_ℓ on the diagonal, and Φ is an $n \times n$ \mathbf{A} -orthogonal matrix whose ℓ th column is the unit-norm (generalized) eigenvector φ_ℓ . The eigenvector matrix Φ is \mathbf{A} -orthogonal, i.e., $\Phi^T \mathbf{A} \Phi = \Phi \mathbf{A} \Phi^T = \mathbf{I}$, where \mathbf{I} is the identity matrix. Moreover, $\Phi^T \mathbf{W} \Phi = \Lambda$. Thus, we have

$$\begin{aligned} \mathbf{W}\Phi &= \mathbf{A}\Phi\Lambda \Rightarrow \mathbf{A}^{-1}\mathbf{W}\Phi\mathbf{A} = \Phi\Lambda\mathbf{A} \Rightarrow \underbrace{\mathbf{A}^{-1}\mathbf{W}}_{\mathbf{L}} \underbrace{\Phi\mathbf{A}\Phi^T}_{\mathbf{I}} \\ &= \Phi\Lambda\mathbf{A}\Phi^T \Rightarrow \mathbf{L} = \Phi\Lambda\Phi^T\mathbf{A}, \end{aligned} \quad (21)$$

since $\mathbf{A}\Phi^T = \Phi^T\mathbf{A}$ because

$$\begin{aligned} \mathbf{A}\Phi^T &= \mathbf{I} \Rightarrow (\Phi^T\mathbf{A})\mathbf{A}\Phi^T = \Phi^T\mathbf{A} \Rightarrow (\Phi^T\mathbf{A}\Phi)\mathbf{A}\Phi^T \\ &= \Phi^T\mathbf{A} \Rightarrow \mathbf{A}\Phi^T = \Phi^T\mathbf{A}. \end{aligned} \quad (22)$$

Fig. 5 a-c 3D horse model colored by $\varphi_2, \varphi_3, \varphi_4$. d-f level sets of $\varphi_2, \varphi_3, \varphi_4$, respectively



Since the first eigenvalue λ_1 is zero, its associated eigenfunction φ_1 is a constant. Denoting by $a = \sum_{i=1}^n a_i$ the total area of the mesh, the first eigenvector φ_1 of the Laplace-Beltrami matrix is then given by

$$\varphi_1 = \left(\frac{1}{\sqrt{a}}, \frac{1}{\sqrt{a}}, \dots, \frac{1}{\sqrt{a}} \right)^T. \quad (23)$$

All the remaining eigenvectors are orthogonal to φ_1 , i.e.,

$$\langle \varphi_\ell, \varphi_1 \rangle_{\mathbf{A}} = \frac{1}{\sqrt{a}} \sum_{i=1}^n a_i \varphi_\ell(i) = \delta_{\ell 1}, \quad \text{for all } \ell = 1, \dots, n. \quad (24)$$

Therefore,

$$\sum_{i=1}^n a_i \varphi_\ell(i) = \sqrt{a} \delta_{\ell 1}, \quad \text{for all } \ell = 1, \dots, n. \quad (25)$$

Since the eigenfunctions are orthonormal, any real-valued function $f : \mathcal{V} \rightarrow \mathbb{R}$ defined on the mesh vertex set can be written in terms of the eigenfunctions as follows:

$$f = \sum_{\ell=1}^n \alpha_\ell \varphi_\ell, \quad \text{where } \alpha_\ell = \langle f, \varphi_\ell \rangle_{\mathbf{A}}. \quad (26)$$

The top row of Fig. 5 shows a 3D horse model colored by the second, third and fourth eigenfunctions, while the bottom row displays the isocontours of these eigenfunctions.

We can use the variational characterizations of the eigenvalues in terms of the Rayleigh-Ritz quotient. That is, the second eigenvalue is given by

$$\lambda_2 = \inf_{f \perp \varphi_1} \frac{\langle f, f \rangle_W}{\langle f, f \rangle_A} = \inf_{f \perp \varphi_1} \frac{\sum_{i \sim j} c_{ij}(f(i) - f(j))^2}{\sum_i a_i f(i)^2}, \quad (27)$$

and φ_2 is its associated eigenfunction. The eigenfunction φ_2 is displayed in Fig. 6a-b, where each vertex v_i is colored by $\varphi_2(v_i)$. The level curves of φ_2 are shown in Fig. 6c-d.

The eigenvalues and eigenfunctions have a nice physical interpretation: the square roots of the eigenvalues $\sqrt{\lambda_\ell}$ are the eigenfrequencies of the membrane, and $\varphi_\ell(i)$ are the corresponding amplitudes at the i th vertex of the mesh. In particular, the second eigenvalue corresponds to the sound we hear the best [48]. In particular, the second eigenvalue λ_2 corresponds to the sound we hear the best. Moreover, Uhlenbeck [42] showed that the eigenfunctions of the LBO are Morse functions on the interior of the domain of the operator. Consequently, this generic property of the eigenfunctions gives rise to constructing their associated skeleton graphs.

Analogous to Fourier harmonics for functions on a circle, the LB eigenfunctions associated to lower eigenvalues correspond to low frequency modes, whereas those associated to higher eigenvalues correspond to high frequency modes that describe the details on the 3D mesh. As shown

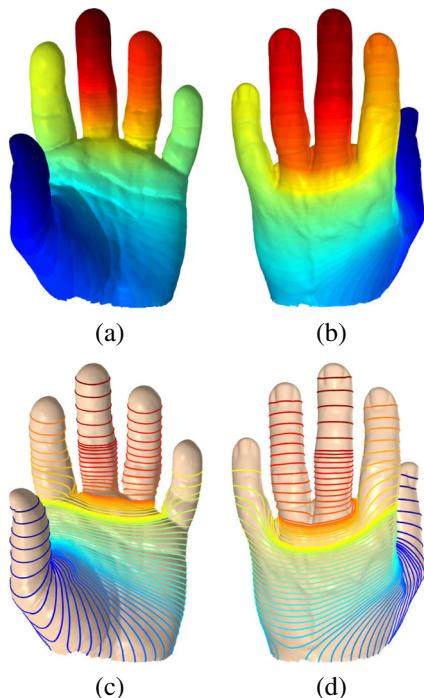


Fig. 6 **a-b** 3D hand model colored by φ_2 (front and back views). **c-d** Level sets of φ_2 (front and back views)

in Figs. 5, 6, and 7a, the second eigenfunction of the LBO captures well the overall shape of 3D objects.

Motivated by the isometric invariance property of the second eigenfunction of the Laplace-Beltrami operator and also by its generic property as a Morse function as well as by the fact that intuitively the second eigenvalue corresponds to the sound we hear the best, we propose to use the spectral Reeb graph to construct the shape skeleton of a 3D object as follows: First, the level sets (isocontours) of the second eigenfunction are computed (identified), as depicted in Fig. 7b, then each level set is encoded as a skeleton node representing the centroid of the isocurve, as shown in Fig. 7c. The main algorithmic steps for computing the spectral Reeb b in detail in Algorithm 1.

Algorithm 1 Proposed skeletonization approach

```

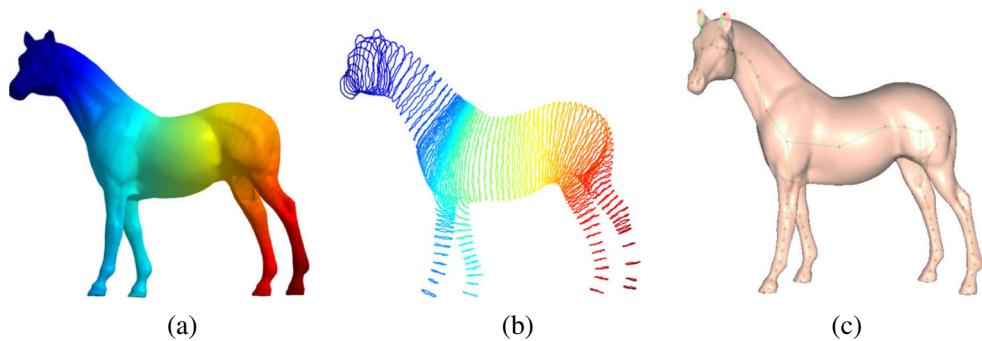
1: Compute the second eigenfunction  $\varphi_2$  of the LBO
   by solving the sparse generalized eigenvalue problem
    $\mathbf{W}\varphi_\ell = \lambda_i \mathbf{A}\varphi_\ell$ .
2: Compute  $N$  level sets  $L_k$  ( $k = 1, \dots, N$ ) of  $\varphi_2$ 
3: for each level set  $L_k$  ( $k = 1$  to  $N$ )
4:   VerticesSetp[0,1] = setIntersect( $\mathbb{M}$ , 1);  $\Leftarrow$  Find vertices (subset of the 3D mesh  $\mathbb{M}$ ) of level set  $L_k$ 
5:   NodeSetp = centroid(VerticesSetp[0,1](n));  $\Leftarrow$ 
      Assign a node to each connected component at its
      centroid.
6: for  $k = 2$  to  $N$ 
7:   VerticesSetc[ $k-1, k$ ] = setIntersect( $\mathbb{M}$ ,  $k-1, k$ );  $\Leftarrow$ 
      Find intersection of  $\mathbb{M}$  from region  $L_{k-1}$  to  $L_k$ 
8:   for each component VerticesSetc [ $k-1, k$ ](n)
9:     NodeSetc = centroid(VerticesSetc [ $k-1, k$ ](n))
10:    for each connected portion
11:      Connect NodeSetc and NodeSetp
12:    end for
13:  end for
14:  NodeSetp = NodeSetc
15:  VerticesSetp = VerticesSetc
16: end for

```

4 Spectral reeb graph matching

In this section, we outline a path similarity skeleton graph matching approach by comparing the relative shortest paths between the spectral skeleton endpoints [39]. The proposed skeleton graph matching is based on the dissimilarity of the shortest paths between the endpoints of the skeletal Reeb graph. A skeleton endpoint refers to the skeleton node that is connected by only one edge as shown in Fig. 8. It is important to point out that endpoints are the salient points of the skeleton and can be seen as visual parts of the original 3D shape [39]. In the same vein as [43], considering only the shortest skeletal paths between endpoints would

Fig. 7 **a** 3D horse model colored by φ_2 ; **b** level sets of φ_2 ; **c** spectral Reeb graph



help avoid the instability problem of the skeleton junction points (i.e. points having three or more adjacent points) and also to make our proposed method more robust to shape deformation. The shortest path between each endpoint and all other endpoints of the skeleton provides an important endpoint feature that will be incorporated into our matching dissimilarity measure.

Our proposed skeleton graph matching approach is based on the assumption that similar skeletons have a similar structure of their endpoints. It is common that the skeletons of similar 3D shapes may have different structures of junction nodes. One of the major advantages of the proposed method is that it does not require that the graphs be converted to trees prior to finding the correspondence, as this conversion may result in the loss of important structural information and, consequently, negatively influence the 3D object recognition result.

In contrast to existing methods for skeleton matching, our proposed approach focuses on the dissimilarity between the shortest paths connecting the skeleton endpoints. We use the shortest paths between endpoints to establish a correspondence relation of the endpoints in different skeletal Reeb graphs. It is worth noting that the idea of using the shortest

paths in skeletal graph matching and classification has been previously explored in the literature. For example, Demirci et al. [44] proposed transforming the graphs into points in a low-dimensional geometric space using low-distortion graph embedding techniques. Each point in the embedding space corresponds to a node in the original graphs. The distance in the embedding space reflects the shortest-path distance in the original graphs in order to keep topological relations. Ling et al. [46] proposed using the inner-distance to build shape descriptors that are robust to articulation and capture part structure. The inner-distance is defined as the length of the shortest path between landmark points within the shape silhouette.

After generating the 3D shape skeleton, our next step is to develop a robust approach for skeletal graph matching. To this end, we match any two Reeb graphs by establishing a correspondence of their endpoints. Then, we apply a pruning algorithm [45] to remove non-salient nodes from the skeleton graph. The proposed matching method consists of two main steps. The first step, which we refer to as indexing, reduces the number of skeletons to be compared with. In the second step, we match the Reeb graphs by applying a dissimilarity measure to retrieve the closest 3D model. These two steps are explained in more details in the following subsections.

4.1 Indexing

A linear search through a database of 3D models is inefficient for large databases, as it requires comparing the query object to each model in the database and selecting the closest one [8]. Therefore, the goal is to apply an efficient indexing mechanism to narrow the search scope in a small set of objects that are most probably similar to the query object. Using our skeletonization algorithm, we may formulate the indexing problem as finding skeletons of which the topological structures are similar to the query skeleton. It is important to note that similar shapes will have the same skeleton even if they are subject to some deformation or transformation. Moreover, these skeletons will have the same number of endpoints.

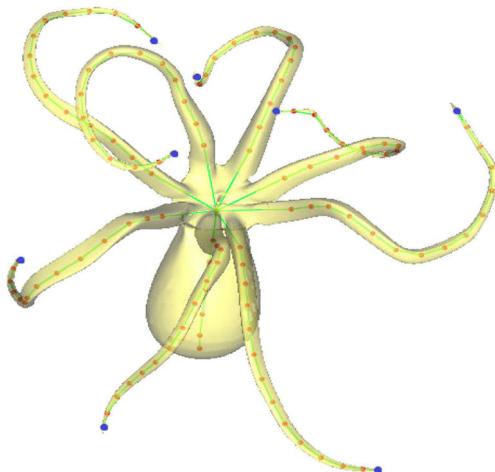


Fig. 8 Spectral Reeb graph of 3D Octopus model and its skeleton endpoints displayed in blue color

Thus, in our indexing mechanism we use the number of skeleton endpoints as the base for indexing, with an error rate of 2 or 3 nodes, meaning that for two skeletons to be in the same index group they should have the same number of endpoints. However, due to noise there might be a difference of 1 or 2 nodes at most, as a result of the pruning process (Fig. 9).

4.2 Endpoint correspondence

After applying the indexing mechanism, the next step is to match the skeletons. Our proposed matching method considers both topological and geometrical features of the matched 3D models. We assign to each endpoint in the Reeb graph (query or model) some features that may help identify the closest endpoint in the other skeletal graph. Thus, our skeleton graph matching problem may be reduced to finding the best correspondence between the endpoints in the query and the endpoints in the model. This can be achieved by minimum weight matching of the two sets of endpoints. A dissimilarity measure between the set of endpoints in both query and model skeletons is used. Therefore, the matching problem aims at finding the best correspondence between the query skeleton endpoints and the database skeletons endpoints. Two endpoints are said to be in close correspondence if the dissimilarity measure between their endpoints has a smaller value. In other words, the matching problem is now reduced to finding the maximum correspondence, minimum weight matching of the two sets of endpoints. The endpoint correspondence process is shown in Algorithm 2.

Algorithm 2 Endpoint correspondence

Let $E = (\mathbf{v}_i)_{i=1,\dots,n_1}$ and $\tilde{E} = (\tilde{\mathbf{v}}_j)_{j=1,\dots,n_2}$ be two sets of endpoints.

For each endpoint $\mathbf{v}_i \in E$:

- 1: Compute a dissimilarity measure between \mathbf{v}_i and all the nodes in \tilde{E}
- 2: Find the node $\tilde{\mathbf{v}}_j$ with the minimum dissimilarity and assign its correspondence to \mathbf{v}_i
- 3: Delete \mathbf{v}_i and $\tilde{\mathbf{v}}_j$ from the list of nodes in E and \tilde{E} , respectively

Repeat steps 1-3 for all nodes in E until one of the node sets E or \tilde{E} is empty

4.3 Matching endpoints using skeleton paths

Endpoint features When generating the skeletal Reeb graph of a 3D shape we assign three features to each endpoint of the skeleton. The first feature is the relative node area, which is equal to the area of the neighboring triangles of the endpoint divided by the total area of the 3D model.

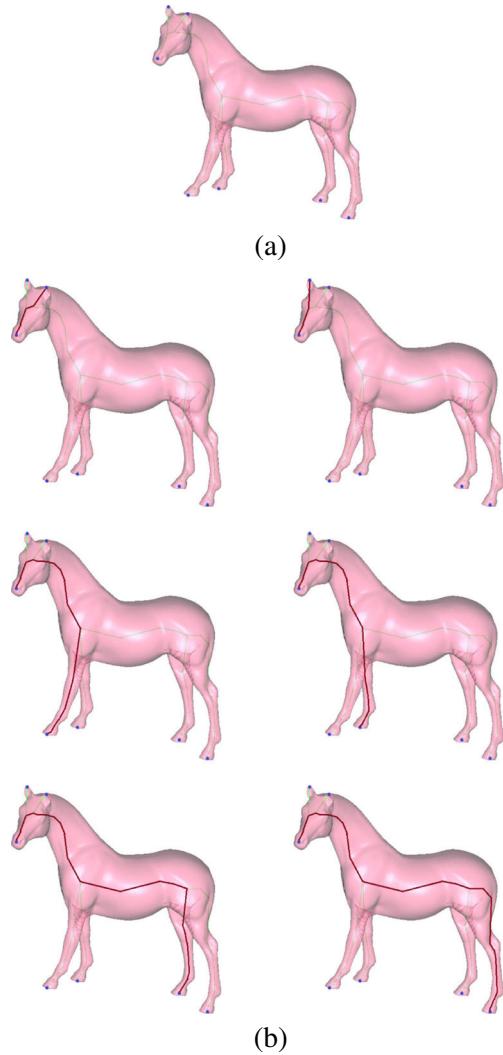


Fig. 9 **a** Horse's spectral Reeb graph. **b** Shortest paths between pairs of endpoints on the skeleton

This feature provides important information about the endpoint as sometimes the skeletons of two models may look similar, albeit their shapes are completely different. Thus, adding this feature to an endpoint will help discriminate between endpoints based on the original 3D shape and not just its skeleton. The reason behind using the relative area is its invariance to scaling. The second feature assigned to an endpoint is the relative node path, which is equal to the sum of shortest path distances from each endpoint to all other endpoints of the skeleton (see Fig. 10b) divided by the sum of the shortest paths from the mesh centroid (root node) to each endpoint. And the third feature is the relative centroid path, which is the shortest path distance from the mesh centroid to each endpoint (see Fig. 10), divided by the sum of the shortest paths from the mesh centroid to all endpoints.

Endpoints dissimilarity Let M and \tilde{M} be two 3D objects with spectral Reeb graphs G and \tilde{G} , respectively. And

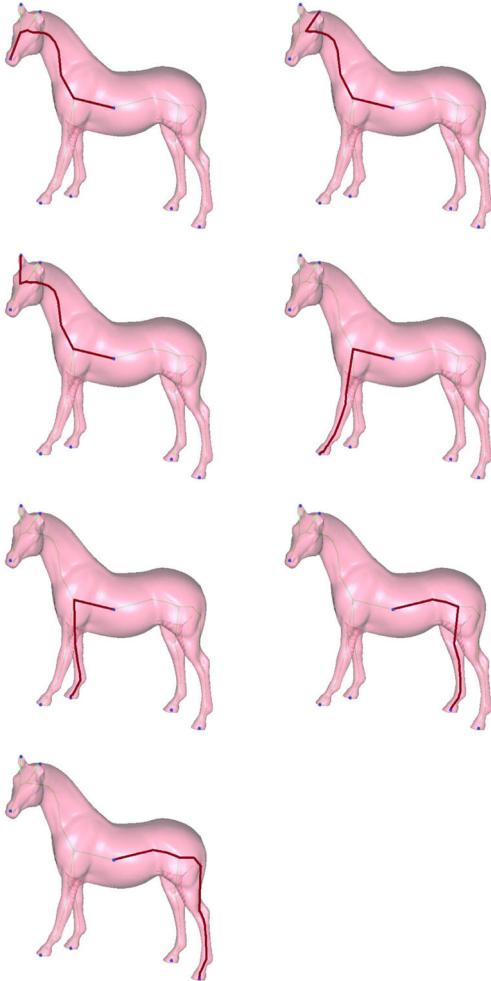


Fig. 10 Shortest paths between the mesh centroid and an endpoint on the skeleton

denote by $E = (\mathbf{v}_i)_{i=1,\dots,n_1}$ and $\tilde{E} = (\tilde{\mathbf{v}}_j)_{j=1,\dots,n_2}$ the skeleton endpoints sets of G and \tilde{G} , respectively. We define the dissimilarity measure between two endpoints \mathbf{v}_i and $\tilde{\mathbf{v}}_j$ as follows:

$$\Phi(\mathbf{v}_i, \tilde{\mathbf{v}}_j) = [(a_i - \tilde{a}_j)^2 + (d\mathbf{v}_i - d\tilde{\mathbf{v}}_j)^2 + (d\mathbf{c}_i - d\tilde{\mathbf{c}}_j)^2]^{1/2}, \quad (28)$$

where

- a_i and \tilde{a}_j are the relative node areas of \mathbf{v}_i and $\tilde{\mathbf{v}}_j$
- $d\mathbf{v}_i = \sum_{k=1}^{n_1} \delta(\mathbf{v}_i, \mathbf{v}_k) / \sum_{k=1}^{n_1} \delta(\mathbf{c}, \mathbf{v}_k)$ and $d\tilde{\mathbf{v}}_j = \sum_{k=1}^{n_2} \delta(\tilde{\mathbf{v}}_j, \tilde{\mathbf{v}}_k) / \sum_{k=1}^{n_2} \delta(\tilde{\mathbf{c}}, \tilde{\mathbf{v}}_k)$ are the relative node paths of \mathbf{v}_i and $\tilde{\mathbf{v}}_j$
- $d\mathbf{c}_i = \delta(\mathbf{c}, \mathbf{v}_i) / \sum_{k=1}^{n_1} \delta(\mathbf{c}, \mathbf{v}_k)$ and $d\tilde{\mathbf{c}}_j = \delta(\tilde{\mathbf{c}}, \tilde{\mathbf{v}}_j) / \sum_{k=1}^{n_2} \delta(\tilde{\mathbf{c}}, \tilde{\mathbf{v}}_k)$ are the relative centroid paths of \mathbf{v}_i and $\tilde{\mathbf{v}}_j$
- \mathbf{c} and $\tilde{\mathbf{c}}$ are the centroids of \mathbb{M} and $\tilde{\mathbb{M}}$, respectively
- $\delta(\cdot, \cdot)$ denotes the Dijkstra's shortest path distance.

Therefore, the dissimilarity between two skeletal Reeb graphs may be defined as:

$$\mathcal{D}(G, \tilde{G}) = \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} \Phi(\mathbf{v}_i, \tilde{\mathbf{v}}_j). \quad (29)$$

The main algorithmic steps of the proposed graph matching approach are described in more detail in Algorithm 3.

Algorithm 3 Proposed graph matching approach

Given two 3D objects \mathbb{M} and $\tilde{\mathbb{M}}$

- 1: Generate the skeletal Reeb graphs G and \tilde{G} of \mathbb{M} and $\tilde{\mathbb{M}}$, respectively
- 2: Apply graph pruning to remove non-salient nodes
- 3: Find the skeleton endpoints sets $E = (\mathbf{v}_i)_{i=1,\dots,n_1}$ and $\tilde{E} = (\tilde{\mathbf{v}}_j)_{j=1,\dots,n_2}$ of G and \tilde{G} , respectively
- 4: **forall** endpoints (\mathbf{v}_i) and $(\tilde{\mathbf{v}}_j)$
- 5: Compute the relative node areas a_i and \tilde{a}_j of \mathbf{v}_i and $\tilde{\mathbf{v}}_j$, respectively
- 6: Compute the relative node paths $d\mathbf{v}_i$ and $d\tilde{\mathbf{v}}_j$
- 7: Compute the relative centroid paths $d\mathbf{c}_i$ and $d\tilde{\mathbf{c}}_j$
- 8: **end For**
- 9: Apply Algorithm 2 to find the correspondence between G and \tilde{G}
- 10: Compute the dissimilarity $\mathcal{D}(G, \tilde{G})$ given by Eq. 29.

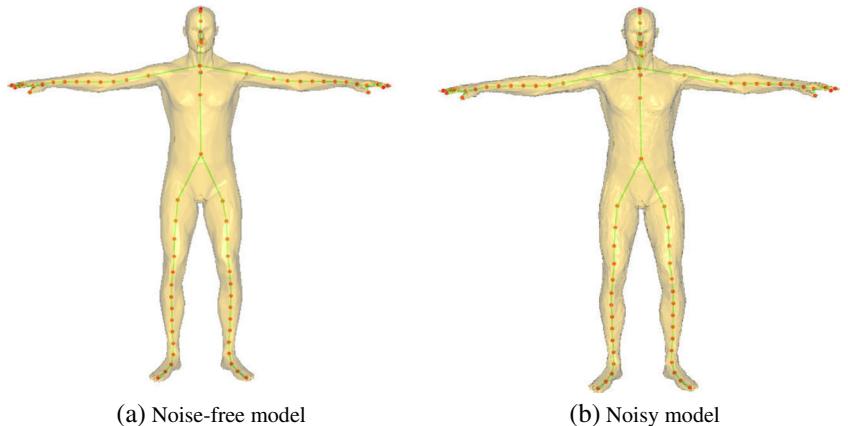
5 Experimental results

In this section, we conduct extensive experiments for 3D shape matching and retrieval to evaluate the performance of the proposed framework. We implemented our algorithms using C++, OpenGL and MATLAB. The experiments were performed on an iMac desktop computer with an Intel Core i5-2400S running at 2.50 GHz and 8 GB RAM.

5.1 Robustness to noise

An important property of any shape descriptor is the ability to match similar shapes even in the presence of noise. To that end, we start by demonstrating the robustness of the proposed skeletonization algorithm to noise. Figure 11 depicts the extracted spectral Reeb graphs of a noise-free model and its noisy version using the proposed algorithm. It is evident that the spectral skeletonization algorithm shows a good preservation of the mesh topological structure. Note that the difference between the skeletons of the original model and the noisy model is minor and not readily noticeable.

Fig. 11 Robustness of spectral Reeb graph to noise



5.2 Matching performance

We tested the performance of the proposed matching algorithm using the McGill Shape Benchmark [8]. This publicly available database provides a 3D shape repository, which contains 255 objects that are divided into ten categories, namely, ‘Ants’, ‘Crabs’, ‘Spectacles’, ‘Hands’, ‘Humans’, ‘Octopuses’, ‘Pliers’, ‘Snakes’, ‘Spiders’, and ‘Teddy Bears’. Sample models from this database are shown in Fig. 12.

The McGill’s database objects are represented by voxel grids as well as by triangle meshes. Table 1 shows that the proposed approach yields correct matching results, where a low value (displayed in boldface with a colored box around it for emphasis) of the dissimilarity measure indicates that the objects are more similar.

5.3 Retrieval performance

We compared our approach with spherical harmonics (SH) [14], medial surfaces (MS) [8], and Reeb graph patch dissimilarity (RGPD) [39]. In general, given a shape query, a retrieval system basically searches a database of shapes and returns an ordered list of responses (also referred to as the ranked list or retrieved set). The retrieved set of shapes is then given by the top k retrieved shapes. In Table 2, the results show that our method achieves better retrieval

results than the spherical harmonics, where the top ten retrieved 3D objects are displayed (top-to-bottom). As can be seen in Table 2, the proposed approach returns correct results, whereas the SH method yields poor retrieval results (columns 2, 4, and 6). Moreover, the proposed algorithm performs slightly better than the RGPD approach.

With respect to a given query, the shapes in a database can be partitioned into four sets: relevant or not, and retrieved or not. To carry out comparison experiments on the entire benchmark of articulated 3D objects, we evaluated the retrieval performance of the proposed approach using the standard information retrieval evaluation measure of precision *versus* recall curve, where

$$\text{precision} = \frac{\text{No. relevant objects retrieved}}{\text{Total No. objects retrieved}} \quad (30)$$

and

$$\text{recall} = \frac{\text{No. relevant objects retrieved}}{\text{Total No. relevant objects in the collection}} \quad (31)$$

In words, precision is the fraction of retrieved shapes that are relevant, whereas recall is the fraction of all relevant shapes retrieved. In a ranked list, we can measure the precision at each recall point. A precision-recall curve is an informative graph that illustrates the tradeoff between

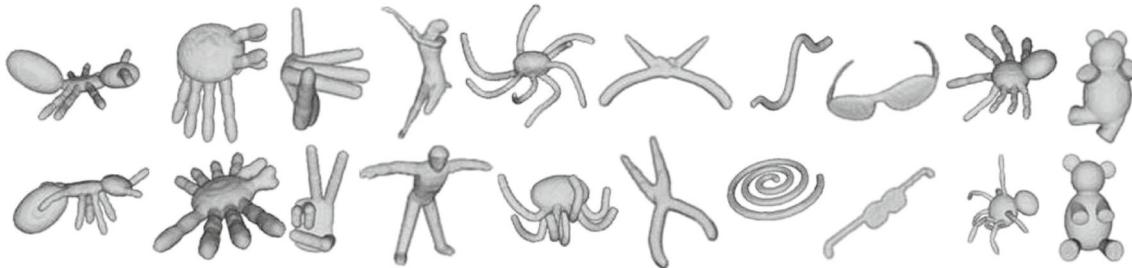


Fig. 12 Sample shapes from McGill’s articulated shape database. Only two shapes for each of the 10 classes are shown

Table 1 Matching results using the proposed approach

	Dinosaur	Bear	Puppy	Dolphin	Hand	Human	X
Dinosaur	0.0124	0.1127	0.1216	0.1258	0.1131	0.1344	0.1257
Bear	0.1116	0.0073	0.1136	0.1297	0.1227	0.1124	0.1131
Puppy	0.1311	0.1142	0.0653	0.1356	0.1315	0.1171	0.1137
Dolphin	0.1146	0.1329	0.1113	0.0055	0.1332	0.1621	0.1552
Hand	0.1193	0.1248	0.1342	0.1421	0.1131	0.1572	0.1592
Human	0.1327	0.1109	0.1152	0.1474	0.11719	0.1021	0.1116
X	0.1223	0.1128	0.1175	0.1453	0.1623	0.1121	0.0042

Each database object is matched against all the other objects in the database. Each cell shows the dissimilarity measure $D(G, eG)$ between two objects selected from the database. The smallest value corresponds to the correct match

precision as a function of recall, and it shows the retrieval performance at each point in the ranking. If, for instance,

Table 2 Retrieval results using the McGill Shape Benchmark

Query			Retrieved Objects								
			SRG	RGPD	SH	SRG	RGPD	SH	SRG	RGPD	SH

The query shapes are shown in the second row. The top ten retrieved objects (top-to-bottom) using spherical harmonics (SH), Reeb graph path dissimilarity (RGPD), and our proposed approach (SRG) are shown in rows 5 to 14

the $(k + 1)$ th shape retrieved is relevant, then both precision and recall increase. However, if it is irrelevant then recall is the same as for the top k shapes, but precision decreases. Hence, a precision-recall curve that is shifted upwards and to the right indicates superior performance. A perfect retrieval result produces a horizontal curve (at precision = 1.0), indicating that all the shapes within the query objects class are returned as the top ranked matches. As can be seen in Fig. 13, our method outperforms spherical harmonics, medial surfaces, and the Reeb graph path dissimilarity approach.

Finally, we tested the performance of the proposed approach on the Princeton Shape Benchmark [19]. The Princeton Shape Benchmark is a publicly available database of 3D polygonal models collected from the Word Wide Web, along with a set of software tools that are widely used by researchers to report their shape matching and retrieval results and compare them to the results of existing algorithms. As can be seen in Table 3, the proposed approach shows superior performance over spherical harmonics, where the top five retrieved 3D objects are displayed (top-to-bottom).

6 Comprehensive comparison of retrieval performance

In this section, we validate the effectiveness of our SRG approach by performing a comprehensive comparison with thirteen recent state-of-the-art methods (see [56] and references therein). The performance of SRG is evaluated on the challenging SHREC 2015, which is a new publicly available 3D shape benchmark consisting of 1200 watertight mesh

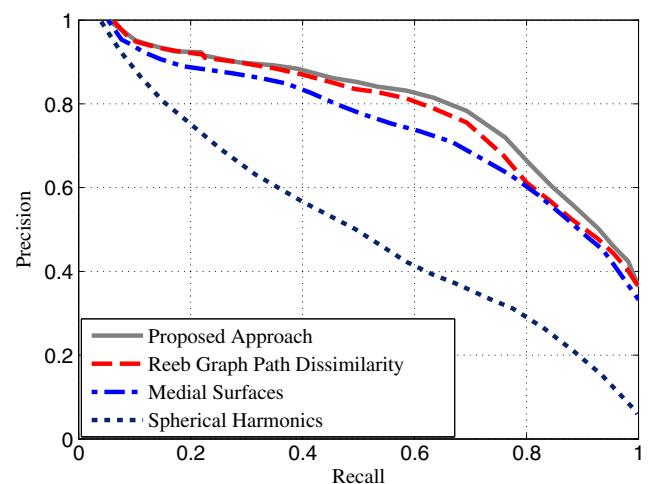


Fig. 13 Precision vs. Recall curves for Reeb graph path dissimilarity, spherical harmonics, medial surfaces, Reeb graph path dissimilarity, and proposed approach using the McGill Shape Benchmark

Table 3 Retrieval results using Princeton 3D Benchmark

Query						
Retrieved Objects						

The query shapes are shown in the second row. The top five retrieved objects (top-to-bottom) of our proposed approach

models that are equally classified into 50 categories. Sample shapes from the SHREC-2015 dataset are shown in Fig. 14.

6.1 Performance evaluation measures

The retrieval performance of SRG is comprehensively evaluated using the following five commonly-used measures that have scores ranging from 0 to 1 (or equivalently from

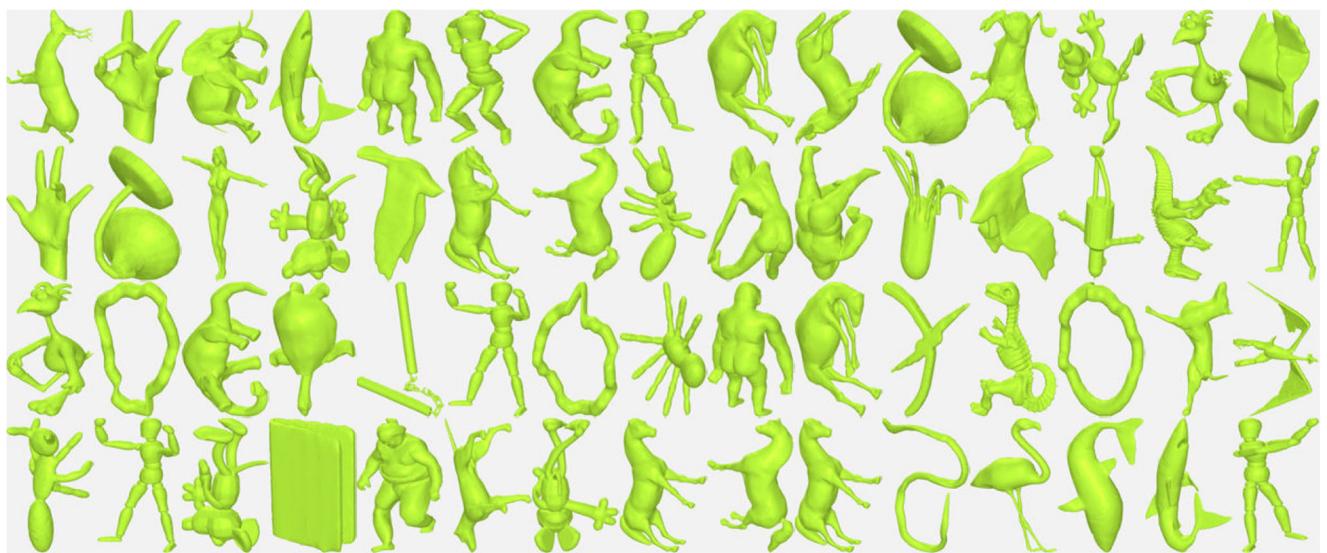
0 % to 100 % in terms of percentages), with a higher score indicating a better performance:

Nearest neighbor (NN), First-Tier (FT), Second-Tier (ST) These three evaluation metrics measure the percentage of shapes in the query's class that appear within the top k retrieved results (i.e. top k matches). If C is the size of the query's class, then $k = 1$ for NN, $k = C - 1$ for FT, and $k = 2(C - 1)$ for ST. In words, the NN metric is the percentage of the closest matches that belong to the query's class, i.e. for each shape in the dataset, the second result (assuming that the first result is the shape itself) is checked whether it is a member of the same class the shape belongs to. The FT metric is the percentage of the $C - 1$ matches retrieved that belong to the query's class, while the ST metric is the percentage of the $2(C - 1)$ matches retrieved that belong to the query's class. The final score of each of these performance metrics is the average over all the shapes in the dataset.

E-measure The E-measure E_k at rank position k is defined as

$$E_k = 1 - \frac{2P_k R_k}{P_k + R_k}, \quad (32)$$

where P_k and R_k are the precision and recall, respectively, at rank position k for a given query. Note that the maximum value E_k is 1, and higher values indicate better results. A user of a search engine is usually more interested in the first page of query results than in later pages. Therefore, we may consider only the first r retrieved shapes for every query and calculates the E-Measure over those results.

**Fig. 14** Sample shapes from the SHREC-2015 dataset

Discounted cumulative gain (DCG): The Discounted Cumulative Gain is a statistic that weights correct results near the front of the list more than correct results later in the ranked list, under the assumption that a user is less likely to consider elements near the end of the list. The DCG_k at a particular rank position k is computed recursively as follows:

$$DCG_k = \begin{cases} G_k & \text{if } k = 1 \\ DCG_{k-1} + \frac{G_k}{\log_2 k} & \text{otherwise} \end{cases} = G_1 + \sum_{\ell=2}^k \frac{G_\ell}{\log_2 \ell}, \quad (33)$$

where G_k is a gain (usefulness) value that depends on the relevance of the k th retrieved shape (1 if the shape that is the k th closest match to the query shape in the query shape's class, and 0 otherwise). The gain is accumulated starting at the top of the ranking and may be reduced, or discounted, at lower ranks. Basically, DCG_k is the total gain accumulated at a particular rank k , and it represents the relevance of the top- k results. Intuitively, the DCG associate a cumulative gain score to each position in the result. The gains, however, are discounted by the logarithm of the position because gains at ranks closer to 1 are generally assumed to be more important than the gains at ranks that are higher. Thus, the DCG score reflects the performance of the algorithm when correct results that are retrieved earlier are weighted higher than those retrieved later. Unlike the average precision which only takes into account binary relevance ratings, the DCG can handle different numbers of relevant levels (i.e. multi-graded relevance). The DCG may have values that can be quite high if there are many highly relevant shapes, and hence it should be normalized. The normalized DCG at rank k is defined as

$$NDCG_k = \frac{1}{Z_k} DCG_k, \quad (34)$$

where Z_k is a normalization factor, called the ideal DCG at position k . The ideal DCG is the maximum achievable DCG at the same rank for a given set of queries, and can be easily found by calculating the DCG of a ranked list that places all the highest-graded shapes above all the second-graded shapes and so on. An explicit cutoff rank k is required to compute DCG and NDCG. All normalized DCG calculations are relative values in the interval $[0, 1]$, making the NDCG more appropriate for averaging over queries. Higher values of NDCG indicate better retrieval performance.

6.2 Retrieval evaluation results

We compared the proposed SRG approach to thirteen 3D shape retrieval methods [56], namely geodesic distance distribution (SNU), heat kernel signature (HKS), surface area

Table 4 Retrieval results (%) on the SHREC-2015 dataset.

Method	Retrieval Evaluation (%)				
	NN	FT	ST	E	DCG
SNU	89.8	56.3	66.9	51.6	83.2
HKS	6.5	6.3	12.4	7.4	39.1
SA	6.5	6.7	12.8	7.8	39.3
WKS	13.4	7.4	13.7	8.3	40.8
Multi-feature	45.0	18.6	26.2	18.4	52.5
SG	93.6	66.8	73.6	58.7	87.5
FVF-HKS	96.0	72.5	80.9	64.4	91.3
FVF-SIHKS	98.0	82.4	88.2	71.7	95.0
FVF-WKS	97.6	82.2	89.4	72.4	95.1
TSAR	81.3	46.3	54.4	42.0	74.9
SID	79.5	48.4	61.4	45.9	77.8
EDBCF-AV	97.5	76.9	86.8	68.9	93.5
RGPD	97.8	77.3	87.1	71.8	94.7
SRG	98.1	83.5	90.6	73.9	95.5

Boldface numbers indicate the best retrieval performance

(SA), wave kernel signature (WKS), multi-feature, spectral geometry (SG), Fisher vector encoding framework (FVF), time series analysis for shape retrieval (TSAR), sphere intersection descriptor (SID), Euclidean distance based canonical forms (EDBCF-AV), and Reeb graph path dissimilarity (RGPD). The performance evaluation results are summarized in Table 4, which shows the retrieval results of thirteen state-of-the-art methods and the proposed framework using the NN, FT, ST, E-measure and normalized DCG metrics. As can be seen, our SRG approach achieves outperforms all the baseline methods. This better performance is in fact consistent with all the retrieval evaluation measures. For example, the NN value for SRG is 98.1 %, while the NN values for the best baseline methods are 97.8 % and 97.5 %, respectively. From the table, we can also see that SRG yields significant improvement of 8.3 % in NN, 27.2 % in FT, 23.7 % in ST, 22.3 % in E and 12.3 % in DCG compared to the SNU method. In addition, SRG outperforms the recent SG approach by 4.5 % in NN, 16.7 % in FT, 17 % in ST, 15.2 % in E and 8 % in DCG.

7 Conclusions

In this paper, we proposed the use of shortest path distance matching algorithm for shape skeletons constructed from the second eigenfunction of the Laplace-Beltrami operator. The better performance of the proposed framework was demonstrated on McGill's articulated shape database compared to spherical harmonics, medial surfaces, and Reeb

graph path dissimilarity. We also validated our method on the Princeton's shape benchmark and showed that the proposed approach gives also satisfactory results for non-articulated shape models. In addition, we performed a comprehensive comparison with thirteen state-of-the-art methods on the SHREC-2015 dataset. Our ongoing research efforts are currently focused on further improving the results by appropriately choosing more discriminatory endpoint features. We also intend to extend our approach to partial matching of 3D shapes.

References

- Grigorishin T, Abdel-Hamid G, Yang YH (1998) Skeletonization: an electrostatic field-based approach. *Pattern Anal Appl*:163–177
- Fomenko AT, Kunii TL (1997) Topological modeling for visualization. Springer-Verlag, Tokyo
- Milnor J, theory Morse (1963) Princeton University Press. New Jersey
- Min P, Kazhdan M, Funkhouser T (2004) A comparison of text and shape matching for retrieval of online 3D models. In: Proc ECDL, pp 209–220
- Siddiqi K, Shokoufandeh A, Dickinson SJ, Zucker SW (1999) Shock graphs and shape matching. *Int Jour Computer Vision* 35(1):13–32
- Hilaga M, Shinagawa Y, Kohmura T, Kunii TL (2001) Topology matching for fully automatic similarity estimation of 3D shapes. In: Proc SIGGRAPH, pp 203–212
- Shinagawa Y, Kunii TL, Kerrosen YL (1991) Surface coding based on Morse theory. *IEEE Comput Graph Appl* 11(5):66–78
- Siddiqi K, Zhang J, Macrini D, Shokoufandeh A, Bouix S, Dickinson S (2008) Retrieving articulated 3-D models using medial surfaces. *Mach Vis Appl* 19(4):261–275
- Cornea ND, Demirci MF, Silver D, Shokoufandeh A, Dickinson S, Kantor PB (2005) 3D object retrieval using many-to-many matching of curve skeletons. In: Proc. Int. Conf. Shape Modeling and Applications, pp 368–373
- Hassouna MS, Farag AA (2009) Variational curve skeletons using gradient vector flow. *IEEE Trans Pattern Anal Mach Intell* 31(12):2257–2274
- Tagliasacchi A, Zhang H, Cohen-Or D (2009) Curve skeleton extraction from incomplete point cloud. *ACM Trans Graph* 28(3)
- Ankerst M, Kastenmüller G, Kriegel H, Seidl T (1999) 3D shape histograms for similarity search and classification in spatial databases. In: Proc. Int. Sympo. Advances in Spatial Databases, pp 207–226
- Osada R, Funkhouser T, Chazelle B, Dobkin D (2002) Shape distributions. *ACM Trans Graph* 21(4):807–832
- Kazhdan M, Funkhouser T, Rusinkiewicz S (2003) Rotation invariant spherical harmonic representation of 3D shape descriptors. In: Proc. ACM Sympo Geometry Processing, pp 156–164
- Ben Hamza A, Krim H (2003) Geodesic object representation and recognitions. In: Proc Discrete Geometry for Computer Imagery, pp 378–387
- Ben Hamza A, Krim H (2006) Geodesic matching of triangulated surfaces. *IEEE Trans Image Process* 15(8):2249–2258
- Ion A, Artner NM, Peyré G, Kropatsch WG, Cohen LD (2011) Matching 2D and 3D articulated shapes using the eccentricity transform. *Comput Vis Image Underst* 115:817–834
- Chen D-Y, Tian X-P, Shen Y-T, Ouhyoung M (2003) On visual similarity based 3D model retrieval. *Comput Graphics Forum* 22(3):223–232
- Shilane P, Min P, Kazhdan M, Funkhouser T (2004) The Princeton shape benchmark. In: Proc Shape Modeling International, pp 167–178
- Ni X, Garland M, Hart JC (2004) Fair morse functions for extracting the topological structure of a surface mesh. In: Proc. Int. Conf. Computer Graphics and Interactive Techniques, pp 613–622
- Tierny J, Vandeborre J-P, Daoudi M (2008) Partial 3D shape retrieval by Reeb pattern unfolding. *Comput Graphics Forum* 28(1):41–55
- Aouada D, Krim H (2010) Squigraphs for fine and compact modeling of 3-D shapes. *IEEE Trans Image Process* 19(2):306–321
- Biasotti S, Giorgi D, Spagnuolo M, Falcidieno B (2007) Reeb graphs for shape analysis and applications. *Theor Comput Sci* 392(1–3):5–22
- Pascucci V, Scorzelli G, Bremer PT, Mascarenhas A (2007) Robust on-line computation of Reeb graphs: simplicity and speed. *ACM Trans Graph* 26(3)
- Patane G, Spagnuolo M, Falcidieno B (2009) A minimal contouring approach to the computation of the Reeb Graph. *IEEE Trans Vis Comput Graph* 15(4):583–595
- Belkin M, Niyogi P, Sindhwan V (2006) Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. *J Mach Learn Res* 7:23992434
- Lévy B (2006) Laplace-Beltrami eigenfunctions: Towards an algorithm that understands geometry. In: Proc. Conf. Shape Modeling and Applications
- Reuter M, Hierarchical shape segmentation and registration via topological features of Laplace-Beltrami eigenfunctions (2010) *Int J Comput Vis* 89(2):287–308
- Rustamov RM (2007) Laplace-Beltrami eigenfunctions for deformation invariant shape representation. In: Proc Symp. on Geometry processing, pp 225–233
- Bronstein AM, Bronstein MM, Guibas L, Ovsjanikov M (2011) Shape Google: Geometric words and expressions for invariant shape retrieval. *ACM Trans Graph* 30(1)
- Tarmissi K, Ben Hamza A (2009) Information-theoretic hashing of 3D objects using spectral graph theory. *Expert Systems with Applications* 36(5):9409–9414
- Sun J, Ovsjanikov M, Guibas L (2009) A concise and provably informative multi-scale signature-based on heat diffusion. *Comput Graphics Forum* 28(5):1383–1392
- Gébal K, Bærentzen JA, Aanæs H, Larsen R (2009) Shape analysis using the auto diffusion function. *Comput Graphics Forum* 28(5):1405–1513
- Shi Y, Lai R, Krishna S, Sicotte N, Dinov I, Toga AW (2008) Anisotropic Laplace-Beltrami eigenmaps: Bridging Reeb graphs and skeletons. In: Proc CVPR Workshops, pp 23–28
- Aubry M, Schlickewei U, Cremers D (2011) The wave kernel signature: A quantum mechanical approach to shape analysis. In: Proc Computational Methods for the Innovative Design of Electrical Devices, pp 1626–1633
- Li C, Ben Hamza A (2013) A multiresolution descriptor for deformable 3D shape retrieval. *Vis Comput* 29(6):513–524
- Zhong M, Qin H (2014) Sparse approximation of 3D shapes via spectral graph wavelets. *Vis Comput* 30(8):751–761
- Li C, Ben Hamza A (2014) Spatially aggregating spectral descriptors for nonrigid 3D shape retrieval: A comparative survey. *Multimedia Systems* 20(3):253281
- Mohamed W, Ben Hamza A (2012) Reeb graph path dissimilarity for 3D object matching and retrieval. *Vis Comput* 28(3):305–318

40. Chung MK, Taylor J (2004) Diffusion smoothing on brain surface via finite element method. In: Proc. IEEE Int. Sympo. Biomedical Imaging: Nano to Macro, pp 432–435
41. Qiu A, Bitouk D, Miller MI (2006) Smooth functional and structural maps on the neocortex via orthonormal Bases of the Laplace-Beltrami operator. *IEEE Trans Med Imaging* 25(10):1296–1306
42. Uhlenbeck K (1976) Generic properties of eigenfunctions. *Am J Math* 98(4):1059–1078
43. Bai X, Latecki LJ (2008) Path similarity skeleton graph matching. *IEEE Trans Pattern Anal Mach Intell* 30(7):1282–1292
44. Demirci MF, Shokoufandeh A, Keselman Y, Bretzner L, Dickinson S (2006) Object recognition as many-to-many feature matching. *Int J Comput Vis* 69(2):203–222
45. Bai X, Latecki LJ, Liu W-Y (2007) Skeleton pruning by contour partitioning with discrete curve evolution. *IEEE Trans Pattern Anal Mach Intell* 29(3):449–462
46. Ling H, Jacobs DW (2007) Shape classification using inner-distance. *IEEE Trans Pattern Anal Mach Intell* 29(2):286–299
47. Rosenberg S (1997) The Laplacian on a Riemannian Manifold. Cambridge University Press
48. Gallot S, Hulin D, Lafontaine J (1990) Riemannian Geometry. Springer-Verlag
49. Wardetzky M, Mathur S, Kälberer F, Grinspun E (2007) Discrete Laplace operators: no free lunch. In: Proc Eurographics Symposium on Geometry processing, pp 33–37
50. Kac M (1966) Can one hear the shape of a drum *Am Math Mon* 73(4):1–23
51. Meyer M, Desbrun M, Schröder P, Barr A (2003) Discrete differential-geometry operators for triangulated 2-manifolds. *Visual Mathematics* III:35–57
52. Kreyszig (1968) Introduction to Differentiable Geometry and Riemannian Geometry. Academic Press
53. Boothby WM (1986) An Introduction to Differentiable Manifolds and Riemannian Geometry. Academic Press
54. Krim H, Ben Hamza A (2015) Geometric methods in signal and image analysis. Cambridge University Press
55. Gordon C, Webb D (1996) You can't hear the shape of a drum. *Am Sci* 84(1):46–55
56. Lian Z, Zhang J, Choi S, ElNaghy H, El-Sana J, Furuya T, Giachetti A, Guler RA, Isaia L, Lai L, Li C, Li H, Limberger FA, Martin R, Nakanishi RU, Neto AP, Nonato LG, Ohbuchi R, Pevzner4 K, Pickup D, Rosin P, Sharf A, Sun L, Sun X, Tari S, Unal G, Wilson RC (2015) SHREC'15 Track: Non-rigid 3D Shape Retrieval. In: Proc Eurographics Workshop on 3D Object Retrieval